ARMY RESEARCH LABORATORY

Clutter Rejection Using Eigenspace Transformation

Lipchen Alex Chan and Nasser M. Nasrabadi

ARL-TR-1997 August 1999

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory

Adelphi, MD 20783-1197

ARL-TR-1997 **August 1999**

Clutter Rejection Using Eigenspace Transformation

Lipchen Alex Chan and Nasser M. Nasrabadi

Sensors and Electron Devices Directorate

Abstract

The goal of our research is to develop an effective and efficient clutter rejector with the use of an eigenspace transformation and a multilayer perception (MLP) that can be incorporated into an automatice target recognition (ATR) system. An eigenspace transformation is used for feature extraction and dimensionality reduction. The transformations considered in this research are principal component analysis (PCA) and the eignespace separation transform (EST). We fed the result of the eigenspace transformation to an MLP that predicts the identity of the input, which is either a target or clutter.

Our proposed clutter rejector was tested on two huge and realistic datasets of second generation forward-looking infrared (FLIR) imagery for the Comanche helicopter. In general, both the PCA and EST methods performed satisfactorily with minor differences. The EST method performed slightly better when a smaller amount of transformed data were fed to the MLP, or when the positive and negative EST eigentargets were used together.

Contents

Executive Summary 1

1 Introduction 3

1.1 Background 3

1.2 Research Objectives 5

2 Eigentargets 7

2.1 Principal Component Analysis 7

2.2 Eigenspace Separation Transform 10

3 Clutter Rejection 13

4 Experimental Results 15

4.1 Manually Produced Chips 15

4.2 Detector-Produced Chips 17

5 Conclusions 24

References 26

Distribution 29

Report Documentation Page 31

Figures

1 Typical processing steps in an ATR system 4

2 An FLIR image taken in a typical environment 5

3 Proposed clutter rejector 6

4 The 100 most dominant PCA eigentargets for targets in a training set 8

5 The 100 (out of 800 possible) most dominant EST eigentargets associated with positive (top) and negative (bottom) eigenvalues for a given training set 12

6 A simple MLP with two layers of weights and shortcut connections 14

7 Examples of SIG images (top two rows) and ROI images (bottom row) 16

8 Examples of clutter images for training (top row) and testing (bottom row) 16

9 Rapid attenuation of eigenvalues in PCA and EST eigentargets 17

10 Examples of target chips (top row) and clutter chips (bottom row) in detector-produced image chips 19

11 First 50 most dominant PCA eigentargets for targets (top five rows) and clutter (bottom five rows) in detector-produced training set 20

12 First 50 most dominant EST eigentargets associated with positive (top five rows) and negative (bottom five rows) eigenvalues for detector-produced training set 20

Tables

1 Performance of PCA method on manually produced chips with various number of MLP inputs 18

2 Performance of EST method on manually produced chips with various number of MLP inputs 18

3 Number of ROI images in second distribution of chips . . . 19

4 Performance of PCA method on the detector-produced dataset with various number of MLP inputs 21

5 Performance of EST method on the detector-produced dataset with various number of MLP inputs 21

6 Performance of PCA method on the detector-produced dataset with equal number of target and clutter eigentargets 22

7 Performance of EST method on the detector-produced dataset with equal number of positive and negative eigentarget . . . 23

Executive Summary

The operation of artilliary and other weapons on the battlefield is often characterized by a series of detection, recognition, tracking, decision-making, and firing processess. Should a mistake occur during any of these activities, the results could be tragic and devastating. Although human operators are usually very good at detecting and recognizing different targets in a range of environments, their performance can be drastically impaired by poor visibility and prolonged operation. As a result, friendly and unintended targets may be accidentally destroyed. Such a tragedy is not uncommon in a military operation. It occurred during Operation Desert Storm in 1991 and has allegedly happened recently in the NATO bombing of Kosovo. Furthermore, certain hostile environments are either inaccessible or too dangerous for human operators to work in. To compensate for such human limitations, an accurate and versatile automatic target recognition (ATR) system is needed. Before something is recognized as a hostile target by an ATR system, its presence and location, which are often overshadowed by a host of distracting but unintended objects or clutter, must be accurately detected. Hence, an effective clutter rejection scheme is needed to single out the potential hostile target from the confusing clutter nearby.

The goal of our research is to develop an effective and efficient clutter rejector with the use of an eigenspace transformation and a multilayer perceptron (MLP). The input to the clutter rejector module is the region of interest (target chip) that is produced by a target detector module within the ATR system. An eigenspace transformation is used for feature extraction and dimensionality reduction. The transformations considered in this research are principal component analysis (PCA) and the eignespace separation transform (EST). These transformations differ in their capabilities to enhance the class separability and to extract compact features from a given training set. The result of the eigenspace transformation is then fed to an MLP that predicts the identity of the input, which is either a target or clutter.

The proposed clutter rejector was tested on two huge and realistic datasets of second-generation forward-looking infrared (FLIR) imagery for the Comanche helicopter. These images were collected at different sites (Ft. Hunter-Ligget, CA; Yuma Proving Ground, AZ; and Camp Grayling, MI), during different months (January, February, June, and August), and at

different times of day (day and night), and for different operational conditions of the target (hot and cold). In the first dataset, the target within each target chip was manually centered. The dataset consists of 10 military targets taken at various sites and viewing aspects. To train our clutter rejector, we used 10,397 signature (SIG) target chips taken with targets in the open, as well as 8,349 competitive clutter chips. For testing, we used 3,456 challenging region-of-interest (ROI) target chips that were taken under poor environmental conditions, and 2,782 clutter chips. Choosing a 10 percent false-alarm rate for our clutter rejector, we managed to detect 98.71 and 92.30 percent of the targets in the first training and testing set, respectively. In the second dataset, the chips were automatically extracted from the ROI scenes by an automatic target detector. Many of these chips contained an off-center target. There were 4,627 target chips and 43,089 clutter chips in the second training set, while another 2,459 target chips and 18,070 clutter chips were used for testing. The best detection rate acheived for the second dataset, controlled at a 10 percent false-alarm rate, was 83.27 and 74.74 percent for the training and testing set, respectively.

Based on the experiments on the first dataset, up to 6 percent of deterioration in detection performance can be attributed to the noisier image quality of the ROI chips. Considering the coupled effect of *unseen surprise* of a typical testing set, the noise factor is not overwhelming in this case. On the other hand, the effect of the off-center targets in the second dataset poses a more severe problem. The testing performance dropped from 92.30 to 74.74 percent, mainly because of this factor. Therefore, a better target-centering algorithm should be developed for the precursory target detector. In general, both the PCA and EST methods performed satisfactorily with minor differences. The EST method performed slightly better when less transformed data were fed to the MLP, or when the positive and negative EST eigentargets were used together.

1. Introduction

1.1 Background

Human beings are usually very good at detecting and recognizing different targets even in relatively crowded and changing environments. However, human performance deteriorates drastically in a low-visibility environment or after an extended period of surveillance. Furthermore, certain working environments are either inaccessible or too hazardous for human beings. To compensate for such human limitations, an accurate and versatile automatic target recognition (ATR) system is needed. For example, an ATR system in a battlefield might alert graveyard-shift sentries with accurate information about any approaching vehicle, so that they could respond quickly.

Unfortunately, the development of such systems is hampered by the large number of target classes and aspects, long viewing ranges, obscured targets, high-clutter backgrounds, different geographic and weather conditions, sensor noise, and variations caused by the translation, rotation, and scaling of the targets. The recognition problem is made even more challenging [1,2] by inconsistencies in the signatures of the targets, similarities between the signatures of different targets, limited training and testing data, camouflaged targets, the nonrepeatability of target signatures, and the deficiency in using any contextual information.

The ATR learning environment in which the training data are collected also exerts a powerful influence on the design and performance of an ATR system. Dasarathy [3] described these environments in an increasing order of difficulty, namely, the supervised, imperfectly supervised, unfamiliar, vicissitudinous, unsupervised, and partially exposed environments. In this report, we assume that our training data come from an unfamiliar environment, where the labels of the training data might be unreliable to a level that is not a priori. For the experiments presented in this report, the input images were obtained by a second-generation forward-looking infrared (FLIR) sensor. For these sensors, the signatures of the targets within the scene are severely affected by rain, fog, and foliage [2]. Clark et al [4] used an information theoretic approach to evaluate the information bound of FLIR images to estimate the best possible performance of any ATR algorithm that uses the given FLIR images as inputs. On the other hand, some FLIR enhancement techniques may be used to preprocess the FLIR input

images. Lo [5] examined six of these techniques and found that a variable threshold zonal filtering technique performed most satisfactorily.

Because of the high dimensionality of input images and the scarcity of training data, it is often necessary to reduce the data dimensionality by transforming the input data into a more compact feature space before the classification process. For instance, Lampinen and Oja [6] subdivided the recognition task into the feature extraction and classification stages. Using a combination of Gabor filters and multilayer self-organizing maps (MSOMs), they mapped the original images to a feature space of reduced dimensionality and complexity. A smaller, supervised subspace network classifier was then used to perform the classification in this feature space. Besides the Gabor filter, the principal component analysis (PCA) [7] and the eigenspace separation transformation (EST) [8] are among the other techniques that have been used for dimensionality reduction in a target recognition task.

A complete ATR system may consist of several algorithmic components, such as preprocessing, detection, segmentation, feature extraction, classification, prioritization, tracking, and aimpoint selection [1]. Among these components, we are particularly interested in the detection-classification modules, which are shown in figure 1. The detection module is certainly one of the most important components, because the whole ATR system will not function properly without an excellent detector. Over the years, a number of detection algorithms have been proposed for ATR systems, such as the virtual agile retina target acquisition and classification (VARTAC) system proposed by Hecht-Nielsen and Zhou [9], the fusion of morphological wavelet transform (MWT) algorithm and Gabor basis function (GBF) detection algorithm proposed by Casasent and Neiberg [10], and the ATR relational template matching (ARTM) algorithm proposed by Kramer et al [11].

False alarms are a common problem for detection algorithms. As shown in figure 2, the boxed areas indicate the potential target that was detected by the ARTM algorithm; however, all but one of these are false alarms. Techniques for reducing false-alarm rates are usually part of the detection algorithm; an example is fusing the output from different detection algorithms, a technique described by Casasent and Neiberg [10].

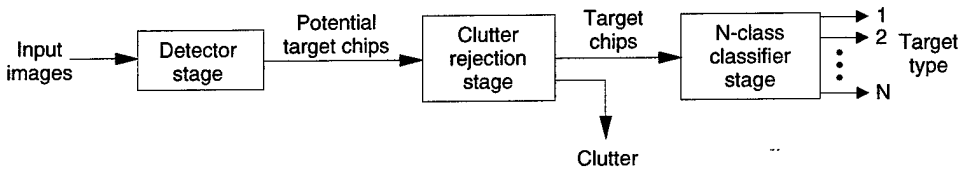


Figure 1. Typical processing steps in an ATR system.

Figure 2. An FLIR image taken in a typical environment. Boxed areas indicate potential targets detected by ARTM algorithm.

1.2 Research Objectives

In a realistic FLIR scenario, such as the one depicted by figure 2, the signatures of certain confusing types of clutter can be very similar to those of a real target. If an automatic target detector has to detect most or all the real targets in the scene, an unacceptable number of false alarms may be produced at the same time. These false alarms could then bog down the performance of the subsequent target classifier in an ATR system.

In this report, we propose a clutter rejector (CR) that effectively reduces the number of false alarms of an automatic target detector operating with the second-generation FLIR imagery. The inputs of this CR are the potential target areas or target chips, similar to those identified with boxes in figure 2. Based on ground-truth information, these chips were labeled as

either a target or clutter. The schematic diagram in figure 3 shows the two stages of our clutter rejector: a set of eigenvectors and a multilayer perceptron (MLP). The eigenvectors or eigentargets, obtained through two different methods, perform feature extraction and dimensionality reduction by transforming the input image chips. The transformed input is then fed to the MLP, where the input is determined as either a target or clutter.

In the next section of this report, we discuss the two eigenspace transformations that we used to construct the eigentargets from the training images. Section 3 describes the neural clutter rejector, which uses the eigentargets as feature templates. Experimental results on two separate distributions of data are presented in section 4. We conclude this report with a brief discussion in section 5.

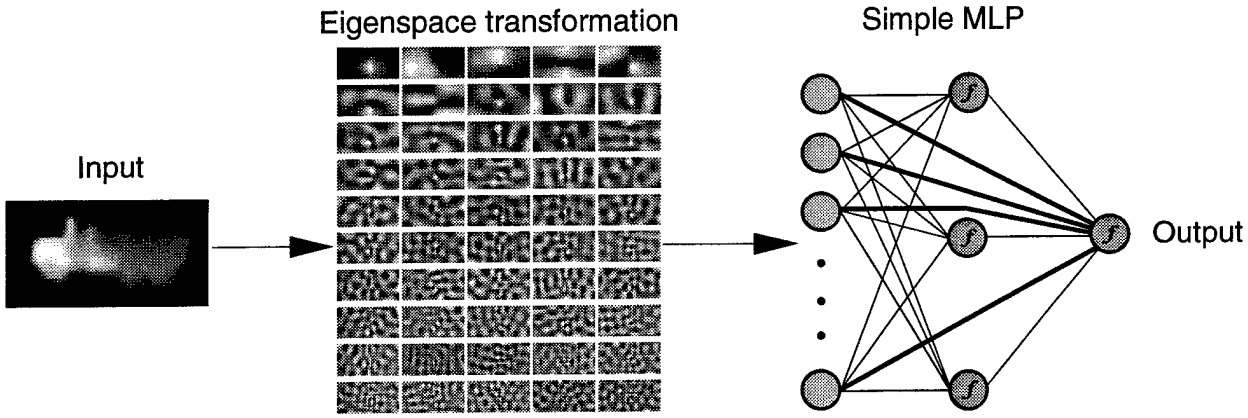


Figure 3. Proposed clutter rejector.

2. Eigentargets

In our experiments, we used two methods to obtain the eigentargets from a given set of training images. Principal component analysis (PCA) is the most basic method, from which the more complicated eigenspace separation transform (EST) method is derived.

2.1 Principal Component Analysis

Also referred to as the Hotelling transform or the discrete Karhunen-Loève transform, PCA is based on statistical properties of vector representations. PCA is an important tool for image processing because it has several useful properties, such as decorrelation of data and compaction of information (energy) [12]. We provide here a summary of the basic theory of PCA. Assume a population of random vectors of the form

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} . \tag{1}$$

The *mean vector* and the *covariance matrix* of the vector population $\mathbf{x}$ are defined as

$$\mathbf{m_x} = E\{\mathbf{x}\} , \text{ and} \tag{2}$$

$$\mathbf{C_x} = E\{(\mathbf{x} - \mathbf{m_x})(\mathbf{x} - \mathbf{m_x})^T\} , \tag{3}$$

where $E\{\arg\}$ is the expected value of the argument, and T indicates vector transposition. Because $\mathbf{x}$ is n-dimensional, $\mathbf{C_x}$ is a matrix of order $n \times n$. Element c_{ii} of $\mathbf{C_x}$ is the variance of x_i (the ith component of the $\mathbf{x}$ vectors in the population), and element c_{ij} of $\mathbf{C_x}$ is the covariance between elements x_i and x_j of these vectors. The matrix $\mathbf{C_x}$ is real and symmetric. If elements x_i and x_j are uncorrelated, their covariance is zero and, therefore, $c_{ij} = c_{ji} = 0$. For N vector samples from a random population, the mean vector and covariance matrix can be approximated from the samples by

$$\mathbf{m_x} = \frac{1}{N} \sum_{p=1}^{N} \mathbf{x}_p , \text{ and} \tag{4}$$

$$\mathbf{C_x} = \frac{1}{N} \sum_{p=1}^{N} (\mathbf{x}_p \mathbf{x}_p^T - \mathbf{m_x} \mathbf{m_x}^T) . \tag{5}$$

Because $\mathbf{C_x}$ is real and symmetric, we can always find a set of n orthonormal eigenvectors for this covariance matrix.

A simple but foolproof algorithm to find these orthonormal eigenvectors for all real symmetric matrices is the Jacobi method [13]. The Jacobi algorithm consists of a sequence of orthogonal similarity transformations. Each transformation is just a plane rotation designed to annihilate one of the off-diagonal matrix elements. Successive transformations undo previously set zeros, but the off-diagonal elements get smaller and smaller, until the matrix is effectively diagonal (to the precision of the computer). We obtain the eigenvectors by accumulating the product of transformations during the process, while the main diagonal elements of the final diagonal matrix are the eigenvalues. Alternatively, a more complicated method based on the QR algorithm for real Hessenberg matrices can be used [13]. This is a more general method because it can extract eigenvectors from a nonsymmetric real matrix. Furthermore, it becomes increasingly more efficient than the Jacobi method as the size of the matrix increases. Given the considerable increase in efficiency for the size of our covariance matrix, we chose the QR method for our experiments described in this report. Figure 4 shows the first 100 (out of the 800 possible in this case) most dominant PCA eigentargets representing the data in the training set. Because they have the largest

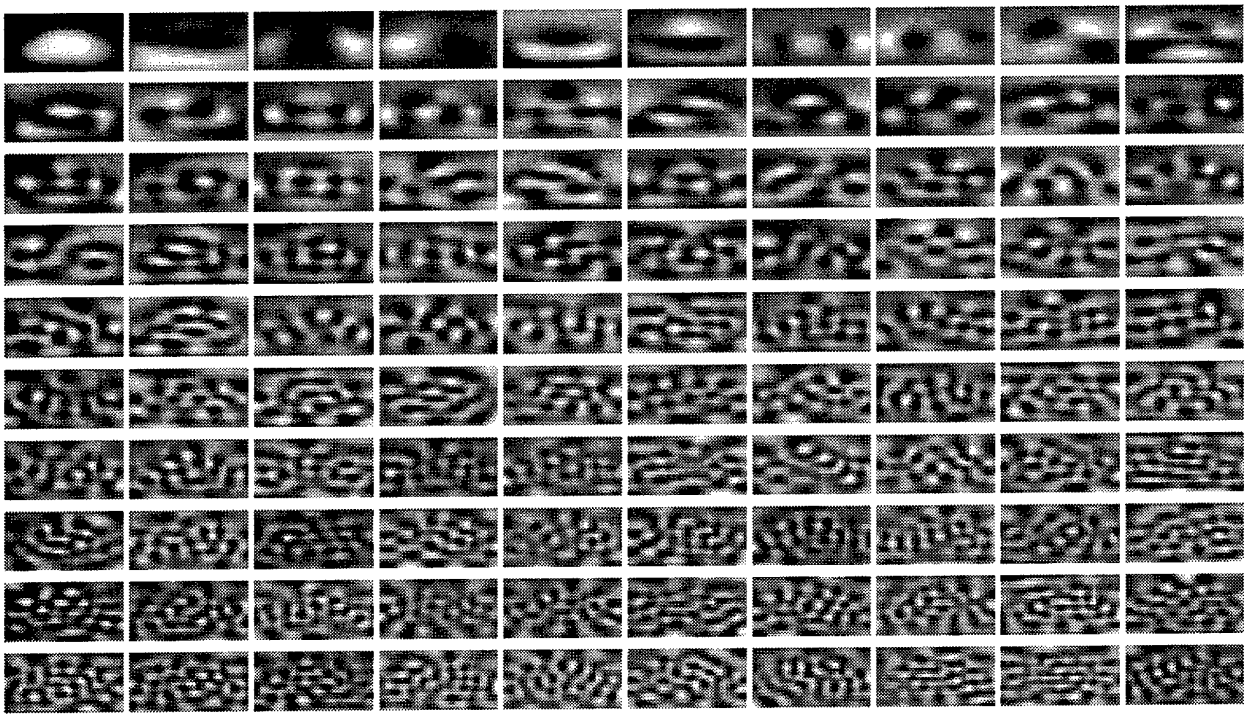


Figure 4. The 100 most dominant PCA eigentargets for targets in a training set.

eigenvalues, these eigentargets capture the greatest variance or energy as well as the most meaningful features among the training data.

Let $\mathbf{e}_i$ and $\lambda_i, i = 1, 2, \ldots, n$, be the eigenvectors and the corresponding eigenvalues of $\mathbf{C_x}$, sorted in a descending order so that $\lambda_j \geq \lambda_{j+1}$ for $j = 1, 2, \ldots, n-1$. Let $\mathbf{A}$ be a matrix whose rows are formed from the eigenvectors of $\mathbf{C_x}$, such that

$$\mathbf{A} = \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \vdots \\ \mathbf{e}_n \end{bmatrix} . \tag{6}$$

This $\mathbf{A}$ matrix can be used as a transformation matrix that maps the $\mathbf{x}$'s into vectors denoted by $\mathbf{y}$'s, as follows:

$$\mathbf{y} = \mathbf{A}(\mathbf{x} - \mathbf{m_x}) . \tag{7}$$

The $\mathbf{y}$ vectors resulting from this transformation have a zero mean vector; that is, $\mathbf{m_y} = \mathbf{0}$. The covariance matrix of the $\mathbf{y}$'s can be computed from $\mathbf{A}$ and $\mathbf{C_x}$ by

$$\mathbf{C_y} = \mathbf{A}\mathbf{C_x}\mathbf{A}^T . \tag{8}$$

Furthermore, $\mathbf{C_y}$ is a diagonal matrix whose elements along the main diagonal are the eigenvalues of $\mathbf{C_x}$; that is,

$$\mathbf{C_y} = \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_n \end{bmatrix} . \tag{9}$$

Because the off-diagonal elements of $\mathbf{C_y}$ are zero, the elements of the $\mathbf{y}$ vectors are uncorrelated. Since the elements along the main diagonal of a diagonal matrix are its eigenvalues, $\mathbf{C_x}$ and $\mathbf{C_y}$ have the same eigenvalues and eigenvectors. In fact, the transformation of the $\mathbf{C_x}$ into $\mathbf{C_y}$ is the essence of the Jacobi algorithm described above.

Therefore, through the PCA transformation, a new coordinate system is established. The origin of this new coordinate system is at the centroid of the population $\mathbf{m_x}$, with new axes in the direction specified by the eigenvectors $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n\}$. The eigenvalue λ_i becomes the variance of component y_i along eigenvector $\mathbf{e}_i$. With its ability to realign unknown data into a new coordinate system based on the principal axes of the data, PCA is often used to achieve rotational invariance in image processing tasks.

However, we may want to reconstruct vector $\mathbf{x}$ from vector $\mathbf{y}$. Because the rows of $\mathbf{A}$ are orthonormal vectors, $\mathbf{A}^{-1} = \mathbf{A}^T$. Therefore, any vector $\mathbf{x}$ can be reconstructed from its corresponding $\mathbf{y}$ by the relation

$$\mathbf{x} = \mathbf{A}^T\mathbf{y} + \mathbf{m}_\mathbf{x} \, . \tag{10}$$

Instead of using all the eigenvectors of $\mathbf{C}_\mathbf{x}$, we may pick only k eigenvectors corresponding to the k largest eigenvalues and form a new transformation matrix $\mathbf{A}_k$ of order $k \times n$. In this case, the resulting $\mathbf{y}$ vectors would be k-dimensional, and the reconstruction given in equation (10) would no longer be exact. The reconstructed vector using $\mathbf{A}_k$ is

$$\hat{\mathbf{x}} = \mathbf{A}_k^T\mathbf{y} + \mathbf{m}_\mathbf{x} \, . \tag{11}$$

The mean square error (MSE) between $\mathbf{x}$ and $\hat{\mathbf{x}}$ can be computed by the expression

$$\epsilon = \sum_{j=1}^{n} \lambda_j - \sum_{j=1}^{k} \lambda_j = \sum_{j=k+1}^{n} \lambda_j \, . \tag{12}$$

Because the λ_j's decrease monotonically, equation (12) shows that we can minimize the error by selecting the k eigenvectors associated with the k largest eigenvalues. Thus, the PCA transform is optimal in the sense that it minimizes the MSE between the vectors $\mathbf{x}$ and their approximations $\hat{\mathbf{x}}$.

2.2 Eigenspace Separation Transform

The EST has been proposed by Torrieri as a preprocessor to a neural binary classifier [8]. The goal of the EST is to transform the input patterns into a set of projection values such that the size of a neural classifier is reduced and its generalization capability is increased. The size of the neural network is reduced because the EST projects an input pattern into an orthogonal subspace of smaller dimensionality. The EST also tends to produce projections with different average lengths for different classes of input and, hence, improves the discriminability between the targets. In short, the EST preserves and enhances the classification information needed by the subsequent classifier. It has been used in a mine-detection task with some success [14].

The transformation matrix $\mathbf{S}$ of the EST can be obtained as follows:

1. Compute the $n \times n$ correlation difference matrix

$$\hat{\mathbf{M}} = \frac{1}{N_1} \sum_{p=1}^{N_1} \mathbf{x}_{1p}\mathbf{x}_{1p}^T - \frac{1}{N_2} \sum_{q=1}^{N_2} \mathbf{x}_{2q}\mathbf{x}_{2q}^T \, , \tag{13}$$

where N_1 and $\mathbf{x}_{1p}$ are the number of patterns and the pth training pattern of Class 1, respectively. N_2 and $\mathbf{x}_{2q}$ are similarly related to Class 2 (which is the complement of Class 1).

2. Calculate the eigenvalues of $\hat{\mathbf{M}}$, $\{\lambda_i \mid i = 1, 2, ..., n\}$.

3. Calculate the sum of the positive eigenvalues

$$E_+ = \sum_{i=1}^{n} \lambda_i \qquad \text{if} \quad \lambda_i > 0\,, \tag{14}$$

and the sum of the absolute values of the negative eigenvalues

$$E_- = \sum_{i=1}^{n} |\lambda_i| \qquad \text{if} \quad \lambda_i < 0\,. \tag{15}$$

(a) If $E_+ > E_-$, then take all the k eigenvectors of $\hat{\mathbf{M}}$ that have positive eigenvalues and form the $n \times k$ matrix $\mathbf{S}$.

(b) If $E_+ < E_-$, then take all the k eigenvectors of $\hat{\mathbf{M}}$ that have negative eigenvalues and form the $n \times k$ matrix $\mathbf{S}$.

(c) If $E_+ = E_-$, then use either subset of eigenvectors to form the matrix $\mathbf{S}$, preferably the smaller subset.

Given the $\mathbf{S}$ transformation matrix, the projection $\mathbf{y}_p$ of an input pattern $\mathbf{x}_p$ is computed as $\mathbf{y}_p = \mathbf{S}^T\mathbf{x}_p$. The $\mathbf{y}_p$, with a smaller dimension (because $k \leq n$) and presumably larger separability between the classes, can then be sent to a neural classifier. Figure 5 shows the eigenvectors associated with the positive and negative eigenvalues of the $\hat{\mathbf{M}}$ matrix that were computed with the target chips as Class 1 and the clutter chips as Class 2. From the upper part of the figure, the signature of targets can be clearly seen. The lower part represents all the features of the clutter.

As we can see from figures 4 and 5, only the first few scores of the eigentargets contain relatively consistent and structurally significant information pertaining to the training data. Yet these eigentargets clearly show a reduction in informational content as their associated eigenvalues decrease. For those less meaningful eigentargets, say from the 50th all the way up to the 800th, only extremely low-intensity and high-frequency information may be present. In other words, by choosing k = 50 in equation (12) when n = 800, the resulting distortion error ϵ would be very small and negligible. While the distortion is negligible, there is a 16-fold reduction in input dimensionality and a similar level of compression in its information content. This property is the essence of eigenspace transformations, and it is very critical to the subsequent satisfactory performance of the neural clutter rejector.

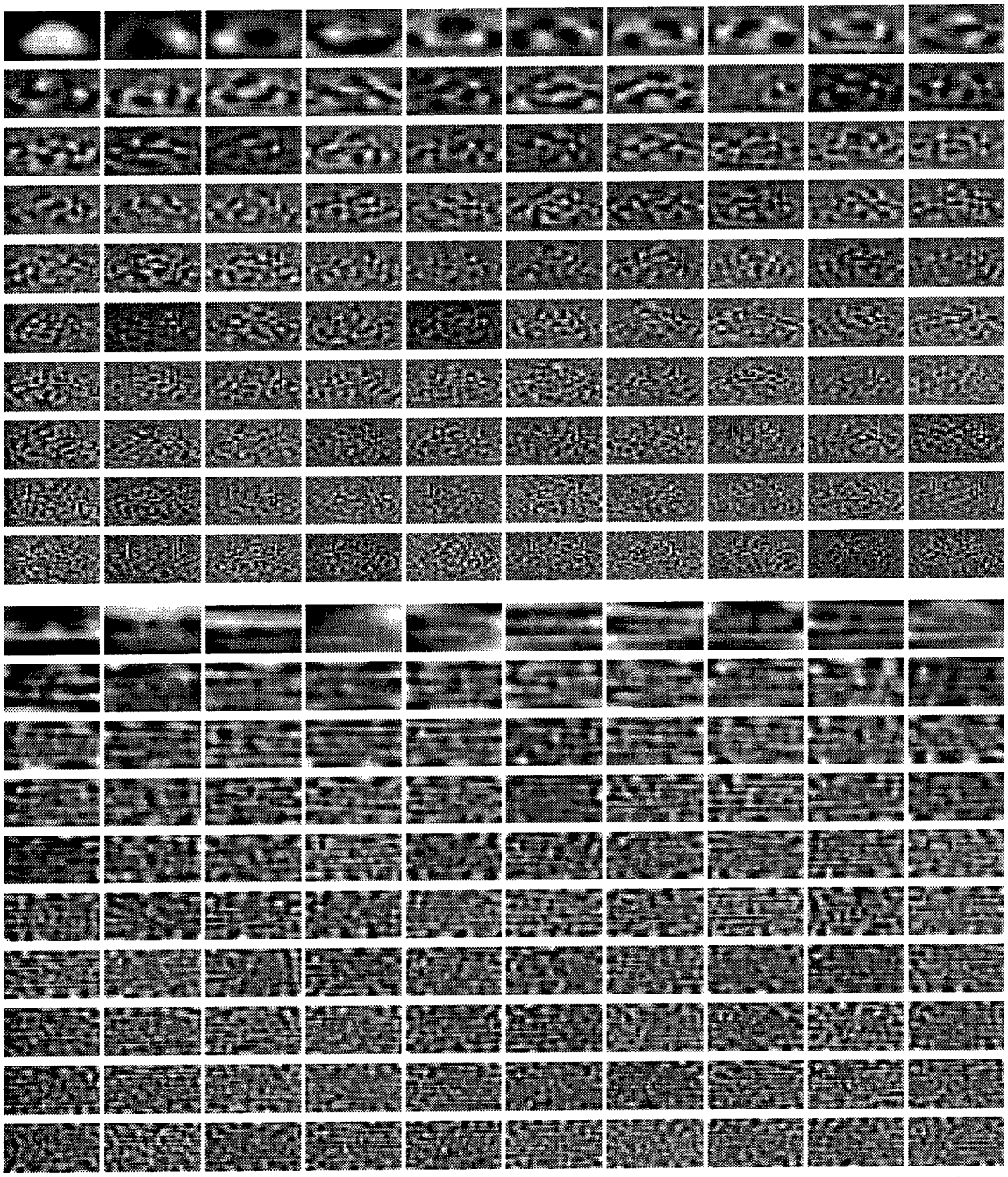


Figure 5. The 100 (out of 800 possible) most dominant EST eigentargets associated with positive (top) and negative (bottom) eigenvalues for a given training set.

3. Clutter Rejection

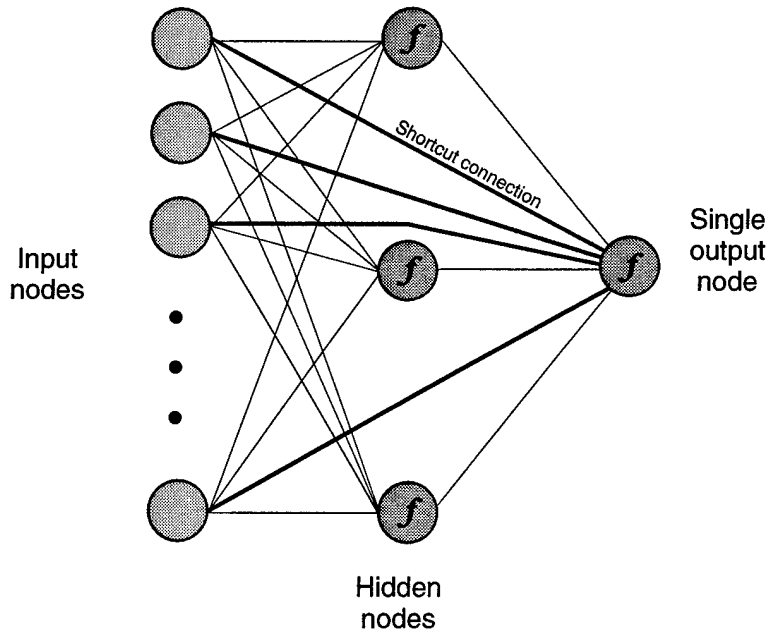
The inputs for our clutter rejection module are the image chips extracted from bigger scenes, as illustrated in figure 2. The size of these image chips is fixed to a predefined dimension, which is common to both the targets and the clutter. To reduce the background information in target chips, we clip each image chip at a size that equals the dimension of the largest target in our training set. After the background removal, the input image is scaled to a preferred size based on a linear interpolation technique. This scaling is needed to achieve an image size that is efficient for feature extraction via the eigenspace transformation, while an effective amount of information is retained in the image.

After normalizing the clipped and scaled training data, we compute the eigentargets using either PCA or the EST. We treat each image pixel as a dimension of the data vector in these computations. The resulting eigentargets are sorted in descending order based on the norm of their corresponding eigenvalues. Characterized by their eigenvalues, different subsets of these eigentargets may be used as feature extractors in different experiments. To achieve feature extraction and dimensionality reduction, we project the preprocessed input image to a chosen set of k eigentargets. The resulting k projection values are fed to an MLP, where they are nonlinearly combined.

A typical MLP used in our experiments is shown in figure 6. The MLP has $k + 1$ input nodes (with an extra bias input), several layers of hidden nodes, and one output node. In addition to full connections between consecutive layers, there are also shortcut connections directly from one layer to all other layers, which may speed up the learning process. The MLP is trained to perform a two-class problem, with training output values of ± 1. Its sole task is to decide whether a given input pattern is a target (indicated by a high output value of around $+1$) or clutter (indicated by a low output value of around -1). The MLP is trained in batch mode by a modified Qprop algorithm [17] for a quick but stable learning course.

If the number of target chips and clutter chips is quite different in the training set, a trained MLP tends to predict the class that has more training samples. This negative effect of an *imbalanced* training set has been studied by Anand et al [18]. To avoid creating such a biased network, we add a corrective measure in our modified learning algorithm. Because the training is

Figure 6. A simple MLP with two layers of weights and shortcut connections.



carried out in batch mode [19], the *error gradient* $\frac{dE}{dw}$ obtained for each network parameter or *weight* for a given training pattern can be accumulated separately, depending on the type of intended outputs for that training pattern. At the end of a training epoch, the average value of the error gradient when the training output is high (low), ε^h (ε^l), for a weight i is computed as

$$\varepsilon_i^h = \frac{1}{N_h} \sum_{p=1}^{N_h} \frac{dE_p^h}{dw_i} \qquad \text{and} \qquad \varepsilon_i^l = \frac{1}{N_l} \sum_{p=1}^{N_l} \frac{dE_p^l}{dw_i}\,, \tag{16}$$

where N_h and N_l are the number of occurrences of high and low training outputs, respectively. If ε_i^h and ε_i^l have the same sign or direction, then their average is used to update the corresponding weight i. Otherwise, no update is made to the controversial weight. This corrective scheme allows the output errors incurred by both high and low target outputs to be reduced simultaneously.

To maximize the class separation between the targets and clutter, we focus only on the training patterns that are easily confused or wrongly classified at a predefined false-alarm rate. Only the errors incurred by these confusing patterns are used to update the MLP weights, so that these patterns may be classified correctly later. A less confusing pattern may be considered only during the early stage of training. This technique of focused learning improves the target recognition rate drastically for a given false-alarm rate.

4. Experimental Results

To examine the performance of our clutter rejection technique, we implement a difficult two-class problem. The input images are 10-bit gray-scale FLIR image chips of both targets and clutter. Similar to the white boxes in figure 2, these chips were extracted with a size of 40×75 pixels from the original image frames. We use two separate distributions of chips, which differ in the way they were extracted. The first distribution of chips was extracted manually based on ground-truth information. The silhouette of a target or clutter is manually centered in each of these chips in most cases, hence, they are relatively easier to recognize. On the other hand, the chips in the second distribution were extracted automatically by a neural automatic target detector (developed at the U.S. Army Research Laboratory (ARL) by Sandor Der and Christopher Dwan). The detected location, instead of the ground-truth location, of a detected target was used to extract a target chip. Since the detected target center is not necessarily the ground-truth center of the target, a lot of target chips in this distribution end up with an off-center target silhouette inside the chip. Similarly, no manual adjustment was made in the extraction of clutter chips in this distribution. Needless to say, these chips are much harder to learn and recognize. Separate experiments were conducted with both distributions of chips, and the results are presented in the following subsections.

4.1 Manually Produced Chips

For the manually produced and silhouette-centered distribution, we have image chips of 10 targets taken at various sites and viewing aspects. Among the target chips, we have a training set of 10,397 SIG (signature) image chips taken with targets in the open. For testing, we use 3,456 challenging ROI (region-of-interest) image chips that were taken under less favorable conditions, such as having targets in and around clutter, in different backgrounds, and under various weather conditions. Typical examples of the SIG and ROI images are shown in figure 7. We have 8,349 and 2,782 clutter image chips for the training and testing purposes, respectively. These clutter images were manually extracted from the same scenery where the SIG and ROI data were obtained. Randomly selected examples of the clutter chips are shown in figure 8. Clearly, some of the clutter images are very similar to the targets in the ROI dataset.

Figure 7. Examples of SIG images (top two rows) and ROI images (bottom row)

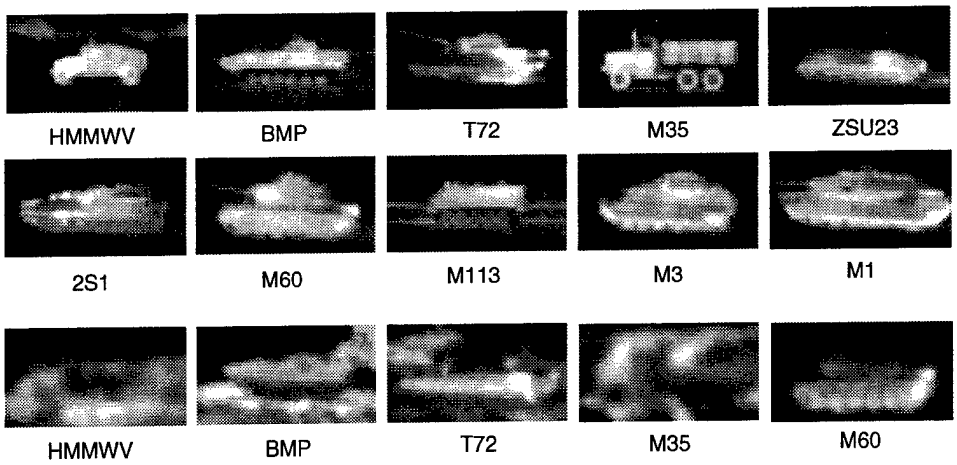
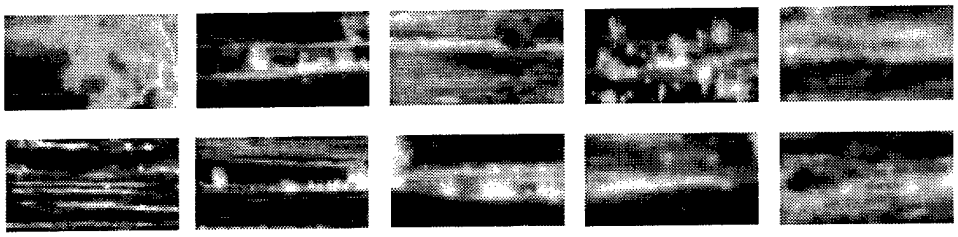


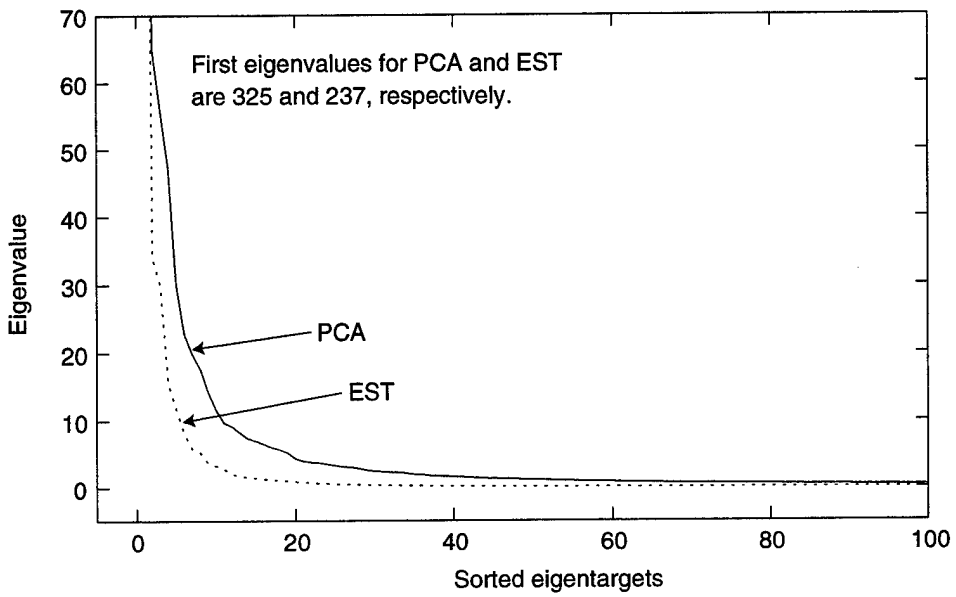
Figure 8. Examples of clutter images for training (top row) and testing (bottom row).



Considering the size of the targets and the computational complexity of the QR algorithm (which is roughly proportional to the cube of the image size), we scale the input image to a moderate size of 40×20 pixels. Using the target chips in the training set, we obtained the corresponding PCA eigentargets as shown in figure 4. Meanwhile, the positive and negative EST eigentargets shown in figure 5 were generated based on both the target and the clutter chips in the training set, in which they form the Class 1 and Class 2 data in equation (13), respectively. We plot the sorted eigenvalues associated with the resulting PCA and positive EST eigentargets, as shown in figure 9. The plots clearly show that the eigenvalues diminish rapidly for both the PCA and EST methods, while those of the EST decrease even faster. In other words, the EST may produce a higher compaction in contextual information. Furthermore, the eigenvalues approach zero after about the fortieth eigentarget. Therefore, we were interested in the 40 most dominant eigentargets only, instead of all 800 eigentargets available.

Theoretically, the more eigentargets employed in the transformation, the larger the amount of information that should be preserved in the transformed data. However, more transformed inputs may quadruple the complexity of the MLP, prolong the training cycle, and increase the chance of getting stuck in a nonoptimal solution. To find the balance between the asymptotically increased information content and the likelihood of

Figure 9. Rapid attenuation of eigenvalues in PCA and EST eigentargets.



obtaining a timely and pseudo-optimal solution, we tried in turn to use the 1, 5, 10, 20, 30, and 40 most dominant eigentargets of each transformation to produce the projection values for the MLP. In each attempt, five independent training processes were tried with different initial random weights for the MLP. The runs were performed at a controlled false-alarm rate of 10 percent. The best performance of each run and the average recognition rate for the five runs are given in tables 1 and 2.

In general, the average recognition rate increases with the number of eigentargets used for feature extraction, but approaches saturation at 30 or more projection values. When fewer projection values are used, significantly higher performance is achieved by the EST. This improvement can be attributed to the better compaction of information associated with EST. However, the testing performance of EST dropped at 40 inputs, which suggests that over-fitted networks may have been created. Furthermore, the slightly lower recognition rates achieved by EST with 20 or more inputs indicate that some minor information might have been lost in this transformation.

4.2 Detector-Produced Chips

We also examined the performance of the proposed CR in the situation where the input chips are both noisy and off-center. In the following experiments, we used the second distribution of image chips, in which the chips were automatically extracted from the challenging ROI image frames by the ARL neural target detector developed by Der and Dwan. In addition

Table 1. Performance of PCA method on manually produced chips with various number of MLP inputs. The false-alarm rate was set at 10 percent.

Number of inputs	Data type	Hit rates (%) and average for five runs					
		1	2	3	4	5	Average
1	Train	67.85	67.85	67.85	67.85	67.85	67.85
	Test	55.01	55.01	55.01	55.01	55.01	55.01
5	Train	88.61	87.88	85.93	87.75	87.91	87.62
	Test	83.42	82.52	80.01	83.62	81.80	82.27
10	Train	96.23	95.96	92.29	93.47	96.33	94.85
	Test	90.13	89.67	86.28	86.81	90.39	88.66
20	Train	97.82	98.25	97.34	97.76	97.32	97.70
	Test	90.08	92.56	89.58	91.15	91.78	91.03
30	Train	97.97	98.43	97.89	98.67	98.02	98.20
	Test	91.09	91.15	90.25	91.90	91.17	91.11
40	Train	99.01	98.72	98.55	98.95	98.32	**98.71**[a]
	Test	93.40	90.71	92.82	92.33	92.25	**92.30**[a]

[a]Bold numbers show discrepency between the performance of the best CR.

Table 2. Performance of EST method on manually produced chips with various number of MLP inputs. The false-alarm rate was set at 10 percent.

Number of inputs	Data type	Hit rates (%) and average for five runs					
		1	2	3	4	5	Average
1	Train	74.35	74.35	74.35	74.35	74.35	74.35
	Test	65.86	65.86	65.86	65.86	65.86	65.86
5	Train	92.68	91.27	92.21	92.11	92.74	92.20
	Test	88.66	87.18	87.99	87.33	88.89	88.01
10	Train	96.19	95.67	96.60	96.55	96.26	**96.25**[a]
	Test	92.36	91.98	92.51	91.52	92.16	**92.11**[a]
20	Train	96.02	96.62	96.14	96.19	94.23	95.84
	Test	90.62	90.74	89.29	91.09	87.76	89.90
30	Train	97.16	96.71	95.05	97.86	94.71	96.30
	Test	91.72	89.87	88.48	91.12	87.27	89.69
40	Train	94.99	94.98	99.30	97.21	95.84	96.46
	Test	86.69	86.89	92.74	90.51	88.02	88.97

[a]Bold numbers show discrepency between the performance of the best CR.

to the noisy nature of these chips, none of them was manually centered during the extraction process. There were only five target types in the ROI database, as identified in the bottom row of figure 7. As shown in table 3, there were 47,716 training image chips in this distribution, in which 4,627 were target chips and 43,089 clutter chips. In the testing set, there were 2,459 target chips and 18,070 clutter chips. The testing set and 29,053 chips

Table 3. Number of ROI images in second distribution of chips.

Data type	Data	Target	Clutter	Total
Training	huli9306	1,049	28,004	29,053
	yuma9202	1,695	10,600	12,295
	gray9201	1,883	4,485	6,368
	Subtotal	4,627	43,089	47,716
Testing	huli9204	2,459	18,070	20,529

of the training set were taken from the same site, but in a different month and year. Randomly selected examples of the target and clutter chips are shown in figure 10. Obviously, some target silhouettes are off-center inside the chip, and the signatures of some clutter chips are very similar to those of the target chips.

Once again, we scaled the input image to a moderate size of 40×20 pixels and generated the PCA and EST eigentargets with this training set. Figure 11 shows the first 50 most dominant PCA eigentargets derived from the target chips (top 5 rows) and clutter (bottom 5 rows) in the detector-produced training set. Compared to figure 4, the features of targets are far less obvious in the top 5 rows of figure 11. Undoubtedly, this is due to the effect of the noisy and off-center target silhouettes in this training set. A similar effect can be observed in the EST eigentargets as well, as shown in figure 12.

As with the manually produced chips, we used the 1, 5, 10, 20, 30, and 40 most dominant eigentargets of each transformation to produce the projection values for the MLP. Similarly, five independent training processes were tried and the performance reported. These results, which were performed at a controlled false-alarm rate of 10 percent, are shown in tables 4 and 5 for the PCA and EST methods, respectively. The performance pattern is quite similar to that expressed in tables 1 and 2, except the detection rates are significantly lower with the detector-produced data. Clearly, the noisy and off-center target silhouettes have created a much tougher learning and recognition task.

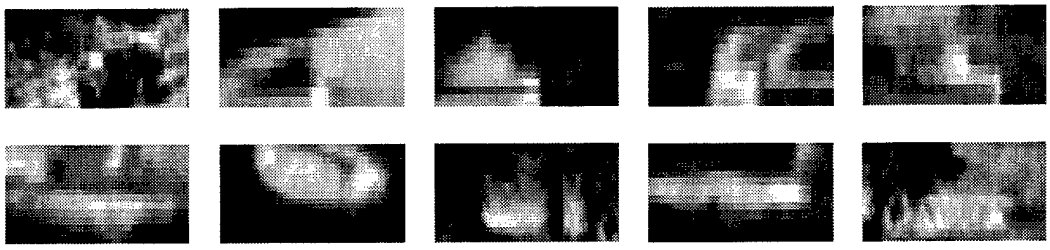


Figure 10. Examples of target chips (top row) and clutter chips (bottom row) in detector-produced image chips.

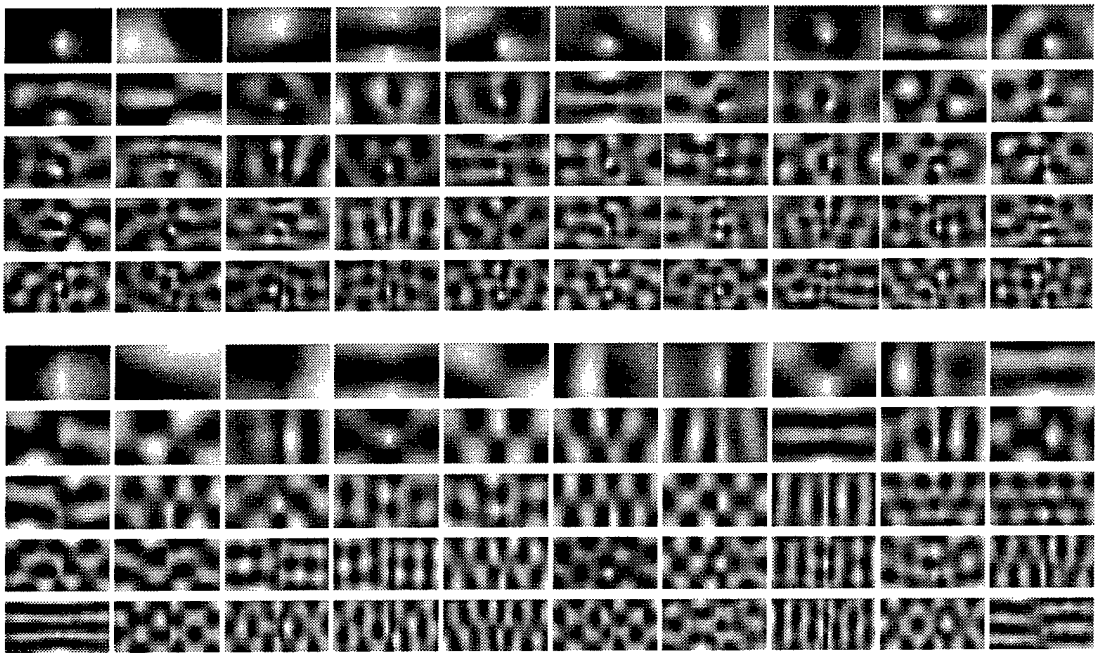


Figure 11. First 50 most dominant PCA eigentargets for targets (top five rows) and clutter (bottom five rows) in detector-produced training set.

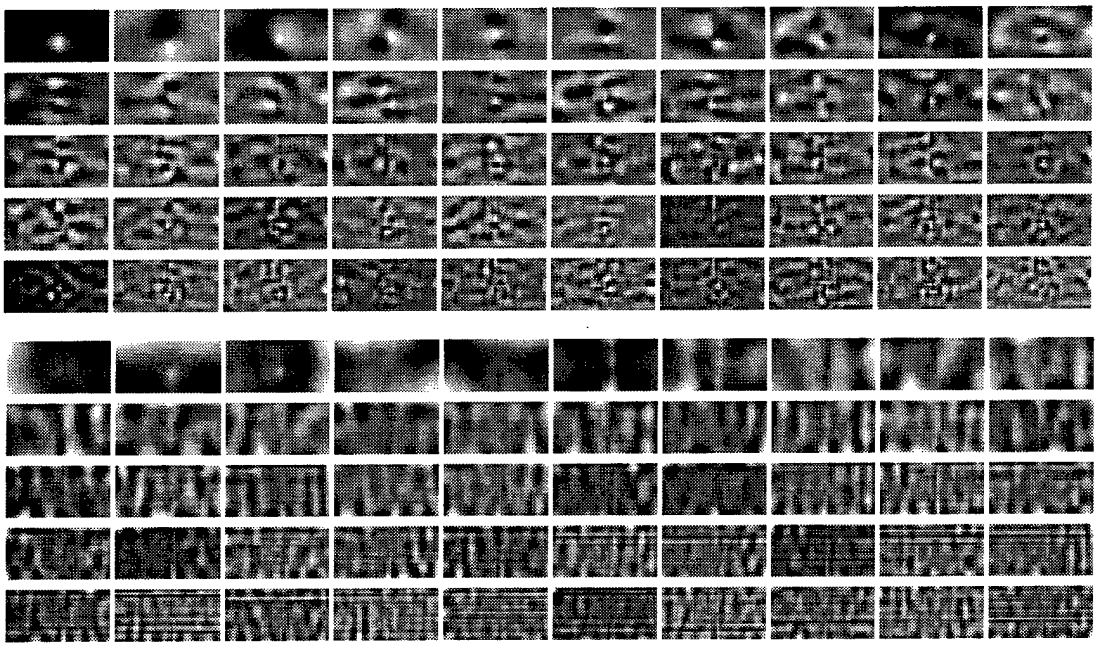


Figure 12. First 50 most dominant EST eigentargets associated with positive (top five rows) and negative (bottom five rows) eigenvalues for detector-produced training set.

Table 4. Performance of PCA method on the detector-produced dataset with various number of MLP inputs. The false-alarm rate was set at 10 percent.

Number of inputs	Data type	Hit rates (%) and average for five runs					
		1	2	3	4	5	Average
1	Train	22.24	22.24	22.24	22.24	22.24	22.24
	Test	28.10	28.10	28.10	28.10	28.10	28.10
5	Train	38.19	38.60	38.60	38.86	37.28	38.31
	Test	45.18	45.47	45.67	45.42	43.07	44.96
10	Train	64.58	63.11	62.05	64.53	63.43	63.54
	Test	64.70	63.40	62.95	66.21	65.51	64.56
20	Train	74.43	76.14	78.56	73.48	76.64	75.85
	Test	70.60	71.00	71.78	70.84	73.24	71.49
30	Train	78.78	82.08	78.71	80.72	83.60	**80.78**[a]
	Test	71.94	73.20	71.86	76.01	73.89	**73.38**[a]
40	Train	81.63	84.37	79.60	85.20	79.90	82.14
	Test	73.69	73.12	69.83	74.79	71.66	72.62

[a]Bold numbers show discripency between the performance of the best CR.

Table 5. Performance of EST method on the detector-produced dataset with various number of MLP inputs. The false-alarm rate was set at 10 percent.

Number of inputs	Data type	Hit rates (%) and average for five runs					
		1	2	3	4	5	Average
1	Train	30.56	30.56	30.56	30.56	30.56	30.56
	Test	30.83	30.83	30.83	30.83	30.83	30.83
5	Train	49.88	50.96	49.54	49.43	50.44	50.05
	Test	52.34	53.64	52.22	52.58	53.31	52.81
10	Train	66.95	65.23	64.69	67.71	66.50	66.22
	Test	65.64	63.97	65.23	67.55	64.95	65.47
20	Train	76.90	75.75	76.72	75.15	76.44	76.19
	Test	72.92	72.10	74.18	72.67	71.78	72.73
30	Train	79.27	79.77	76.31	77.39	77.31	**78.01**[a]
	Test	74.14	74.38	73.04	70.80	71.86	**72.84**[a]
40	Train	74.69	70.41	79.71	75.92	71.28	74.40
	Test	66.25	66.73	72.79	65.72	64.70	67.23

[a]Bold numbers show discripency between the performance of the best CR.

Instead of using only the target PCA and positive EST eigentargets (the top five rows of figures 11 and 12), we also examined the usefulness of clutter PCA and negative EST eigentargets (the bottom five rows of figures 11 and 12). We formed each MLP input pattern by cascading an equal number of projection values produced by the target (positive) and clutter (negative) eigentargets of the PCA (EST). Taking 5 to 30 projections from each side, we

constructed the corresponding MLPs with 10 to 60 inputs. The runs were performed at a controlled false-alarm rate of 10 percent. The best performance of each run and the average recognition rate for the five runs are given in tables 6 and 7.

Based on tables 4 and 6, the clutter PCA eigentargets could improve the hit rates when the number of MLP inputs is small. With 30 or more projections from the target eigentargets, however, the benefit of clutter eigentargets has completely vanished. On the other hand, as shown in tables 5 and 7, the negative EST eigentargets seem to be useful even for the MLPs with 60 inputs. Indeed, the MLPs trained with 30 positive and 30 negative EST projections have achieved the best average performance for the detector-produced dataset. It is possible that the orthonormal relationship between the positive and the negative EST eigentargets has captured more unique and less redundant information in the training data; hence, it enables the subsequent MLP to learn and perform more optimally.

Table 6. Performance of PCA method on the detector-produced dataset with equal number of target and clutter eigentargets. The false-alarm rate was set at 10 percent.

Number of inputs	Data type	Hit rates (%) and average for five runs					
		1	2	3	4	5	Average
10	Train	64.58	62.57	63.84	63.97	64.15	63.82
	Test	66.90	61.77	63.68	65.39	64.25	64.40
20	Train	73.81	73.50	73.68	74.30	70.74	73.21
	Test	73.57	72.55	72.39	72.10	70.64	72.25
30	Train	75.30	76.94	77.65	76.79	70.09	75.35
	Test	73.12	73.93	74.26	74.79	68.85	72.99
40	Train	76.08	80.61	80.16	77.91	78.97	78.75
	Test	73.65	76.25	74.30	73.57	71.94	73.94
50	Train	76.01	74.13	84.03	83.90	82.65	**80.14**[a]
	Test	72.75	69.66	76.49	75.15	74.50	**73.71**[a]
60	Train	83.66	81.87	82.82	75.36	77.09	80.16
	Test	78.00	74.34	74.54	69.46	69.58	73.18

[a]Bold numbers show discripency between the performance of the best CR.

Table 7. Performance of EST method on the detector-produced dataset with equal number of positive and negative eigentargets. The false-alarm rate was set at 10 percent.

Number of inputs	Data type	Hit rates (%) and average for five runs					
		1	2	3	4	5	Average
10	Train	66.48	69.01	68.64	66.20	69.38	67.94
	Test	66.73	70.88	71.61	69.05	69.95	69.64
20	Train	76.59	80.46	77.52	77.89	78.13	78.12
	Test	74.14	74.14	72.88	76.62	76.66	74.89
30	Train	79.06	81.72	82.43	81.05	82.84	81.42
	Test	74.75	75.03	76.62	75.48	77.35	75.85
40	Train	79.08	86.64	86.28	81.33	81.05	82.88
	Test	72.18	77.47	77.63	73.49	71.82	74.52
50	Train	83.99	78.47	84.46	83.99	82.49	82.67
	Test	73.44	70.84	74.83	73.44	74.66	73.44
60	Train	80.42	84.50	82.80	85.80	82.82	**83.27**[a]
	Test	71.13	77.23	75.84	75.68	73.81	**74.74**[a]

[a]Bold numbers show discripency between the performance of the best CR.

5. Conclusions

In this report, we have described a clutter rejection technique that is based on eigenspace transformation and an MLP classifier. The proposed CR is tested with a wide range of realistic FLIR images, in which as many as 10 military targets were viewed from 72 different aspects at several proving grounds and under various meteorological conditions. From the experimental results, a few conclusions can be drawn here.

Based on the performance given in tables 1 and 2, we may say that the effect of noisy testing data is significant but not overwhelming. Due to the factor of *unseen surprise*, the performance of the testing set is usually lower than that of the training set, even when both the training and testing data were collected under the same conditions. From tables 1 and 2, the discrepancy between the training and testing performance of the best CR (which are highlighted with bold typeface) is about 4 to 6 percent. Therefore, the effect of noise introduced by the degraded ROI images should be equal or less than 6 percent, which is not too bad.

The effect of off-center silhouettes, on the other hand, has more severe impacts on the performance of the CR. For instance, the best performance in table 4 is about 16 to 19 percent lower than the corresponding training and testing performance achieved in table 1. This difference can be attributed mainly to the off-center characteristic of the chips, but not to the fact that the ROI (instead of the SIG) images were used as the training set in the case of table 4. Because the experimental results documented in our previous technical report [20] showed that when the manually centered ROI chips were included in the training set, they were able to improve the recognition rate of the ROI chips in the testing data from 75.1 to 91.0 percent. In other words, learning to recognize the ROI testing chips based on an ROI training set alone should not have incurred the 16 to 19 percent drop in performance here. To address this problem, therefore, we need an automatic target detector that can detect the target center more accurately.

Despite the small differences, both the PCA and EST methods have performed satisfactorily in our experiments. Although it is simpler than the EST, the PCA method seems to perform better as a CR when 20 or more projection values are fed to the corresponding MLP. Nonetheless, the EST proves to be a better transformation when only a small number of projection values can be processed, because of speed or memory constraints. Furthermore, the best performance can be achieved when both positive and

negative EST eigentargets are used simultaneously. Using both the target and clutter PCA eigentargets at the same time, on the other hand, does not pose any improvement over having the target PCA eigentargets alone.

Using a dynamically selected active training set to update the MLP weights has been shown to be an effective scheme. By focusing mainly on the training patterns around the region of confusion, the MLP has learned the delicate boundary between the target and clutter more successfully. In our initial experiments, we have noticed an improvement of up to 20 percent in recognition rate due to this dynamic pattern selection process.

References

1. B. Bhanu, "Automatic Target Recognition: State of the Art Survey," *IEEE Transactions on Aerospace and Electronic Systems* **22**, 4 (July 1986), pp 364–379.

2. B. Bhanu and T. Jones, "Image Understanding Research for Automatic Target Recognition," *IEEE Aerospace and Electronic System Magazine* (1993), pp 15–22.

3. B. Dasarathy, "Information Processing for Target Recognition from Autonomous Vehicles," *Proceedings of the SPIE Electro-Optical Technical Autonomous Vehicles*, **219** (1980), pp 86–93.

4. L. Clark, L. Perlovsky, W. Schoendorf, C. Plum, and T. Keller, "Evaluation of Forward-Looking Infrared Sensors for Automatic Target Recognition Using an Information-Theoretic Approach," *Optical Engineering*, **31**, 12 (1992), pp 2618–2627.

5. C. Lo, "Forward-Looking Infrared (FLIR) Image Enhancement for the Automatic Target Cuer System," *Proceedings of the SPIE Image Processing for Missile Guidance*, **238** (1980), pp 91–102.

6. J. Lampinen and E. Oja, "Distortion Tolerant Pattern Recognition Based on Self-Organizing Feature Extraction," *IEEE Transactions on Neural Networks*, **6**, 3 (May 1995), pp 539–547.

7. A. O'Toole, H. Abdi, K. Deffenbacher, and D. Valentin, "Low-Dimensional Representation of Faces in Higher Dimensions of the Face Space," *Optical Society of America*, **10**, 3 (1993), pp 405–411.

8. D. Torrieri, "A Linear Transform That Simplifies and Improves Neural Network Classifiers," *Proceedings of the International Conference on Neural Networks*, **3** (1996), pp 1738–1743.

9. R. Hecht-Nielsen and Y. Zhou, "VARTAC: A Foveal Active Vision ATR System," *Neural Networks*, **8**, 7/8 (1995), pp 1309–1321.

10. D. Casasent and L. Neiberg, "Classifier and Shift-Invariant Automatic Target Recognition Neural Networks," *Neural Networks* **8**, 7/8 (1995), pp 1117–1129.

11. A. Kramer, D. Perschbacher, R. Johnston, and T. Kipp, "Relational Template Matching Algorithm for FLIR Automatic Target Recognition," *Proceedings of the SPIE Architecture, Hardware, and FLIR Issues in ATR*, **1957** (1993), pp 29–37.

12. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Addison-Wesley Publishing, New York (1992).

13. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, 2nd ed., Cambridge University Press, New York (1992).

14. G. L. Plett, T. Doi, and D. Torrieri, "Mine Detection Using Scattering Parameters and an Artificial Neural Network," *IEEE Transactions on Neural Networks*, **8**, 6 (1997), pp 1456–1467.

15. P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**, 7 (1997), pp 711–720.

16. K. Etemad and R. Chellappa, "Dimensionality Reduction of Multiscale Feature Spaces Using a Separability Criterion," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (1995), pp 2547–2550.

17. S. Fahlman, "Faster Learning Variations on Back-Propagation: An Empirical Study," *Proceedings of the 1988 Connectionist Models Summer School*, Morgan Kaufmann (1988), pp 38–51.

18. R. Anand, K. Mehrotra, C. Mohan, and S. Ranka, "An Improved Algorithm for Neural Network Classification of Imbalanced Training Sets," *IEEE Transactions on Neural Networks*, **4**, 6 (1993), pp 962–969.

19. S. Haykin, *Neural Networks: A Comprehensive Foundation*, Macmillan College Publishing, New York (1994).

20. L. A. Chan and N. M. Nasrabadi, *Automatic Target Recognition Using Wavelet-Based Vector Quantization*, U.S. Army Research Laboratory, ARL-TR-1503 (December 1997).

Distribution

Admnstr
Defns Techl Info Ctr
Attn DTIC-OCP
8725 John J Kingman Rd Ste 0944
FT Belvoir VA 22060-6218

Ofc of the Secy of Defns
Attn ODDRE (R&AT)
The Pentagon
Washington DC 20301-3080

OSD
Attn OUSD(A&T)/ODDR&E(R) R J Trew
Washington DC 20301-7100

AMCOM MRDEC
Attn AMSMI-RD W C McCorkle
Redstone Arsenal AL 35898-5240

Army Rsrch Ofc
Attn AMXRO-GS Bach
PO Box 12211
Research Triangle Park NC 27709

CECOM
Attn PM GPS COL S Young
FT Monmouth NJ 07703

Dept of the Army (OASA) RDA
Attn SARD-PT R Saunders
103 Army
Washington DC 20301-0103

Dir for MANPRINT
Ofc of the Deputy Chief of Staff for Prsnnl
Attn J Hiller
The Pentagon Rm 2C733
Washington DC 20301-0300

TECOM
Attn AMSTE-CL
Aberdeen Proving Ground MD 21005-5057

US Army Armament Rsrch Dev & Engrg Ctr
Attn AMSTA-AR-TD M Fisette
Bldg 1
Picatinny Arsenal NJ 07806-5000

US Army CECOM Night Vision & Elec Sensors Dir
Attn B Redman
Attn D N Barr
Attn AMSEL-RD-NV-VISPD C Hoover
Attn AMSRL-RD-NV-UAB C Walters
10221 Burbeck Rd Ste 430
FT Belvoir VA 22060-5806

US Army Edgewood RDEC
Attn SCBRD-TD G Resnick
Aberdeen Proving Ground MD 21010-5423

US Army Info Sys Engrg Cmnd
Attn ASQB-OTD F Jenia
FT Huachuca AZ 85613-5300

US Army Natick RDEC Acting Techl Dir
Attn SSCNC-T P Brandler
Natick MA 01760-5002

Director
US Army Rsrch Ofc
4300 S Miami Blvd
Research Triangle Park NC 27709

US Army Simulation, Train, & Instrmntn Cmnd
Attn J Stahl
12350 Research Parkway
Orlando FL 32826-3726

US Army Tank-Automtv Cmnd Rsrch, Dev, & Engrg Ctr
Attn AMSTA-TA J Chapin
Attn AMSTA-TR-S G Gerhart
Warren MI 48397-5000

US Army Train & Doctrine Cmnd Battle Lab Integration & Techl Dirctrt
Attn ATCD-B J A Klevecz
FT Monroe VA 23651-5850

US Military Academy
Mathematical Sci Ctr of Excellence
Attn MDN-A MAJ M D Phillips
Dept of Mathematical Sci Thayer Hall
West Point NY 10996-1786

Distribution (cont'd)

Nav Surface Warfare Ctr
Attn Code B07 J Pennella
17320 Dahlgren Rd Bldg 1470 Rm 1101
Dahlgren VA 22448-5100

DARPA
Attn B Kaspar
3701 N Fairfax Dr
Arlington VA 22203-1714

Univ of Maryland Dept of Elec Engrg
Attn Q Zheng
Attn R Chellappa
A V Williams Bldg Rm 2365
College Park MD 20742-3285

ERIM
Attn P Rauss
Attn C Dwan
Attn J Ackenhusen
Attn Q Holmes
1975 Green Rd
Ann Arbor MI 48105

Hicks & Associates Inc
Attn G Singley III
1710 Goodrich Dr Ste 1300
McLean VA 22102

Lockheed Martin Vought
Attn B Evans, Stop L37-01
Attn D DuBois, Stop LOST-20
Attn K Jenkins, Stop L37-01
PO Box 650003
Dallas TX 75265-0003

Palisades Inst for Rsrch Svc Inc
Attn E Carr
1745 Jefferson Davis Hwy Ste 500
Arlington VA 22202-3402

Sanders Lockheed Martin Co
Attn PTP2-A001 K Damour
PO Box 868
Nashua NH 03061-0868

US Army Rsrch Lab
Attn AMSRL-DD J Rocchio
Attn AMSRL-CI-LL Techl Lib (3 copies)
Attn AMSRL-CS-AS Mail & Records Mgmt
Attn AMSRL-CS-EA-TP Techl Pub (3 copies)
Attn AMSRL-P-P J M Miller
Attn AMSRL-SE J Pellegrino
Attn AMSRL-SE-EE Z G Sztankay
Attn AMSRL-SE-S A Sindoris
Attn AMSRL-SE-S J Eicke
Attn AMSRL-SE-SA M Wellman
Attn AMSRL-SE-SE L A Chan (10 copies)
Attn AMSRL-SE-SE M Lander
Attn AMSRL-SE-SE M Vrabel
Attn AMSRL-SE-SE N Nasrabadi (5 copies)
Attn AMSRL-SE-SE S Der
Attn AMSRL-SE-R T Kipp
Attn AMSRL-SE-SE H Moon
Attn AMSRL-SE-SE H Kwon
Attn AMSRL-SE-SE B Weber
Attn AMSRL-SE-SE G Stolovy
Attn AMSRL-SE-SE J Penn
Attn AMSRL-SE-SE P Gillespie
Adelphi MD 20783-1197

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	August 1999	Interium October 1998 to April 1999

4. TITLE AND SUBTITLE Clutter Rejection Using Eigenspace Transformation

5. FUNDING NUMBERS
DA PR:A305
PE: 61102A

6. AUTHOR(S) Lipchen Alex Chan and Nasser M. Nasrabadi

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
Attn: AMSRL- SE-SE email: nnasraba@arl.mil
2800 Powder Mill Road
Adelphi, MD 20783-1197

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-1997

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
2800 Powder Mill Road
Adelphi, MD 20783-1197

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
ARL PR: 9NEOMM
AMS code:611102.305

12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The goal of our research is to develop an effective and efficient clutter rejector with the use of an eigenspace transformation and a multilayer perception (MLP) that can be incorporated into an automatice target recognition (ATR) system. An eigenspace transformation is used for feature extraction and dimensionality reduction. The transformations considered in this research are principal component analysis (PCA) and the eignespace separation transform (EST). We fed the result of the eigenspace transformation to an MLP that predicts the identity of the input, which is either a target or clutter.

Our proposed clutter rejector was tested on two huge and realistic datasets of second generation forward-looking infrared (FLIR) imagery for the Comanche helicopter. In general, both the PCA and EST methods performed satisfactorily with minor differences. The EST method performed slightly better when a smaller amount of transformed data were fed to the MLP, or when the positive and negative EST eigentargets were used together.

14. SUBJECT TERMS
ATR, pincipal component analysis, eigenspace separation transformation (EST)

15. NUMBER OF PAGES
38

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102